

Simulation FDZ	Praktikum Software Entwicklung
Anforderungsanalyse Spezifikation	26.03.2018

Systemweite Anforderungsanalyse

**Fabrik der Zukunft
Simulation des Transports**
Andreas Glaser, Dzianis Brysiuk, Noah Lehmann

Simulation FDZ	Praktikum Software Entwicklung
Anforderungsanalyse Spezifikation	26.03.2018

Inhaltsverzeichnis

1. Einleitung.....	3
2. Systemweite funktionale Anforderungen.....	3
2.1 Was implementiert wird.....	3
2.2 Worauf bauen wir auf.....	3
2.3 Was nicht implementiert wird.....	4
3. Systemqualitäten.....	4
3.1 Usability.....	4
3.2 Verfügbarkeit.....	5
3.3 Performance.....	5
3.4 Supportfähigkeit.....	5
4. System Schnittstellen/Interfaces.....	6
4.1 User Interfaces.....	6
4.1.1 Look & Feel.....	6
4.1.2 Layout und Navigationsanforderungen.....	6
4.2 Schnittstellen zu externen Systemen und Geräten.....	8
4.2.1 Software Schnittstellen.....	8
4.2.2 Hardware Schnittstellen.....	8
4.2.3 Kommunikationsschnittstellen.....	8
5. Vorhandenes System.....	8
5.1 Befehle.....	8
5.2 Stationen.....	8
5.3 Schlitten / Palleten.....	8
5.4 Datenformat.....	9
6. System Randbedingungen.....	9
7. System Compliance.....	9

Simulation FDZ	Praktikum Software Entwicklung
Anforderungsanalyse Spezifikation	26.03.2018

1. Einleitung

In dieser Anforderungsanalyse (Lastenheft) werden die Systemweiten Anforderungen genau beschrieben, um das Verständnis der vom Kunden gewünschten Anforderungen zu validieren und die Übereinstimmung mit den Erwartungen der Stakeholder sicherzustellen.

2. Systemweite funktionale Anforderungen

2.1 Was implementiert wird

Das System stellt eine grafische Oberfläche dar, die das Transportwesen einer Fabrik simuliert. Sie bildet also eine Benutzeroberfläche, die die Kontrolle des Transports ermöglicht.

Das System wird lokal auf einem Rechner laufen. Es bekommt über eine Netzwerkschnittstelle eine Nachricht in Textformat, die unabhängig vom laufenden Betrieb simuliert werden kann und dann interpretiert und verarbeitet werden muss. Der Zustand, den die Nachricht beschreibt, wird dann grafisch für den Benutzer dargestellt. Es muss unterschieden werden, ob die Nachricht einfach dargestellt werden kann oder ob sie abhängig vom aktuellen Zustand des Systems individuell interpretiert werden muss. Der Zustand des Systems ist nur abhängig von der Nachricht, die eintrifft, nicht vom echten Betrieb der Fabrik. Jede Handlung und jede Nachricht, die das System vermerkt, muss in einem Log dargestellt und nochmal separat abgespeichert werden. Der Log besteht aus einem Datum und der aktuellen Meldung.

2.2 Worauf bauen wir auf

Das aktuelle Programm der Fabrik versendet über eine Netzwerkschnittstelle Nachrichten an die Transportsimulation. Diese empfängt die Nachrichten aktuell und reagiert dementsprechend. Unser System übernimmt die Netzwerkschnittstelle und interpretiert die Nachrichten genauso. Lediglich die Funktion der Oberfläche sowie der Logs wird neu entworfen.

Simulation FDZ	Praktikum Software Entwicklung
Anforderungsanalyse Spezifikation	26.03.2018

2.3 Was nicht implementiert wird

Das System erfordert keine Authentifizierung des Nutzers.

Außerdem müssen keinerlei Lizenzen beachtet werden, da die Vorliegende Struktur der Fabriksimulation dem Auftraggeber unterliegen. Es sollte allerdings dokumentiert werden, welche Teile übernommen werden.

Auch die Lizenzierung von Software muss nicht beachtet werden, da das System nur mit Open-Source Software entwickelt wird.

Der Zugriff auf Ressourcen lokal auf dem Rechner sowie über das Netzwerk muss nicht gesichert werden.

Es werden auch keine anderen Medien außer eines Standard Rechners unterstützt.

3. Systemqualitäten

3.1 Usability

Das System wird nach Fertigstellung nur von den Entwicklern der „Fabrik der Zukunft“ (Prof. Dr. Plenk, Wissenschaftliche Mitarbeiter), Studenten (mit und ohne Programmierkenntnisse) sowie in Zukunft auch von Industriepartnern genutzt werden. Um das fertige Produkt in Betrieb zu nehmen erfordert es keine weiteren Einstellungen, es wird auf den aktuellen Zustand der Fabrik eingestimmt sein.

Die Ein-/ Ausgabe des Systems erfolgt über die Standard Geräte eines Rechners, Maus, Tastatur und einem Bildschirm.

Die grafische Oberfläche ermöglicht es dem Anwender, die Geschwindigkeit der Nachrichtenverarbeitung des Systems einzustellen, also die Geschwindigkeit, wie Reaktionen auf eine Nachricht optisch dargestellt werden. Außerdem wird die Möglichkeit gegeben, dem Transportband neue Elemente hinzuzufügen. Der Anwender hat außerdem die Möglichkeit, das aktuelle Vorgehen des Systems über Statusmeldungen und einem Log nachzuvollziehen. Dieser Log soll dem Anwender die original Nachricht, sowie eine Interpretation dieser darstellen.

Simulation FDZ	Praktikum Software Entwicklung
Anforderungsanalyse Spezifikation	26.03.2018

3.2 Verfügbarkeit

Das System hat keine feste Betriebszeit, es soll also jederzeit gestartet und auch beendet werden können. Außerdem soll die Beendigung des Systems kontrolliert ablaufen, die Meldungen sollen in einem Log abgespeichert werden, welches täglich neu erstellt wird und unter dem Erstellungsdatum abgespeichert wird. Wird das Programm mehrmals täglich gestartet, so sollen spätere Logs in der selben Datei gespeichert werden, bis diese Datei eine gewisse Speichergröße überschreitet. Die Größe der Datei wird noch von den Entwicklern bestimmt. Die Log-Dateien sollen dem Anwender bei Neustart oder Absturz des Systems zur Verfügung stehen, er soll die Möglichkeit haben, den alten Zustand des Systems zu laden oder es in einen Anfangszustand zu bringen. Dies geschieht durch einen kurzen Abfragedialog (Ja/Nein).

Die Wartbarkeit des Systems ist durch die Dokumentation und die Einhaltung von Entwurfsmustern gegeben.

Das System soll in der Lage sein, ungültige Nachrichten zu verarbeiten, bzw. zu ignorieren, ohne abzustürzen (siehe DocumentationNetworkCommands.pdf, Kapitel 4.1.1). Bei Absturz sollen die Log-Daten abgespeichert sein.

3.3 Performance

Es sind keine Rechenzeit-Anforderungen an das System gestellt. Die einzige Anforderung ist, dass das System die Geschehnisse in zwei Geschwindigkeiten darstellen kann; einmal in Rechenzeit, quasi nicht lesbar für das menschliche Auge und einmal in Realzeit der Fabrik, also so schnell, wie die Paletten tatsächlich verarbeitet werden. Die Realzeit der Fabrik soll vom Anwender einstellbar sein. Es soll konfiguriert werden können, wie lange der Transport eines Schlittens von einer Station zu einer anderen braucht.

Das System soll in der Lage sein, mehrere Schlitten gleichzeitig zu verarbeiten. Die theoretische Obergrenze ist hier dann unendlich, auch wenn höhere Zahlen (mehr als 2) nicht vorgesehen sind.

Das System muss nur ein Transportband simulieren, weitere sind nicht im Umfang dieses Projektes.

Der Anwender muss in der Lage sein, das System jederzeit ohne Weiteres zu starten und zu beenden.

Simulation FDZ	Praktikum Software Entwicklung
Anforderungsanalyse Spezifikation	26.03.2018

3.4 Supportfähigkeit

Die Anpassungsfähigkeit an andere Betriebssysteme genauer zu betrachten ist nicht explizit notwendig, da das System in Java programmiert wird.

Auch die Unterstützung mehrerer Eingabeprotokolle ist nicht notwendig, da die Nachrichten in einem Protokoll verschickt werden, das speziell für die Fabrik entwickelt wurde.

Es werden keine zukünftige Größenzuwächse erwartet, die das System bewältigen muss. Es soll lediglich die Möglichkeit betrachtet werden, neue Geräte in die Struktur des Transportes aufzunehmen.

4. System Schnittstellen/Interfaces

4.1 User Interfaces

4.1.1 Look & Feel

In System werden folgende Inhalte dargestellt:

- Verbindungsstatus zum Übergeordnete Schnittstelle (Transport-Controller)
- Status und Log-Ausgaben
- Simulation von Stationen (RO → Roboter , EA → Ein-/Ausgabe, LA → Lager)
- Simulation von Schlitten mit/ohne Palette mit der zugehörigen ID in den Stationen oder auf dem Transportband
- Stau von Schlitten/Paletten.

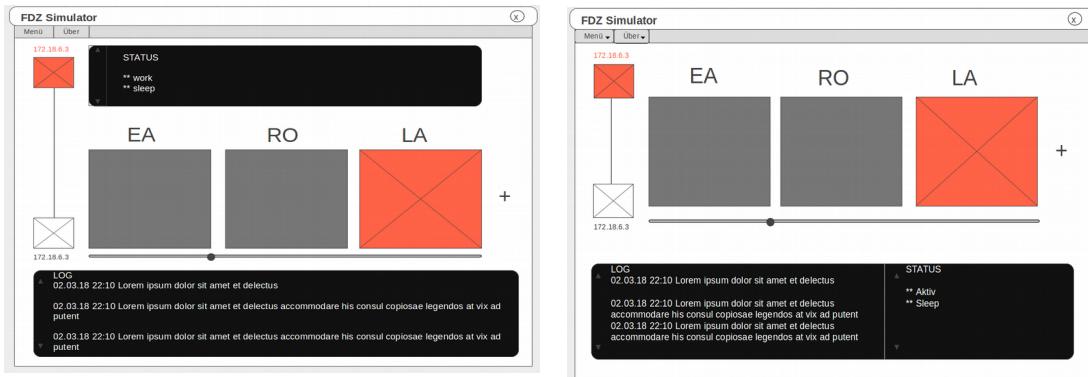
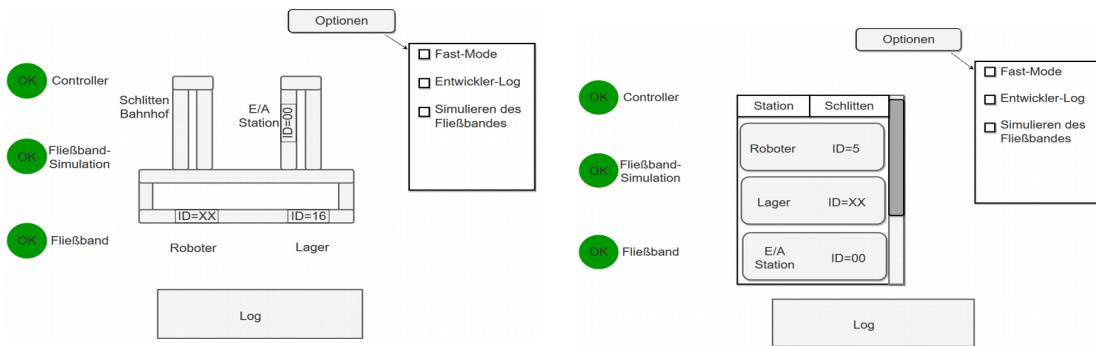
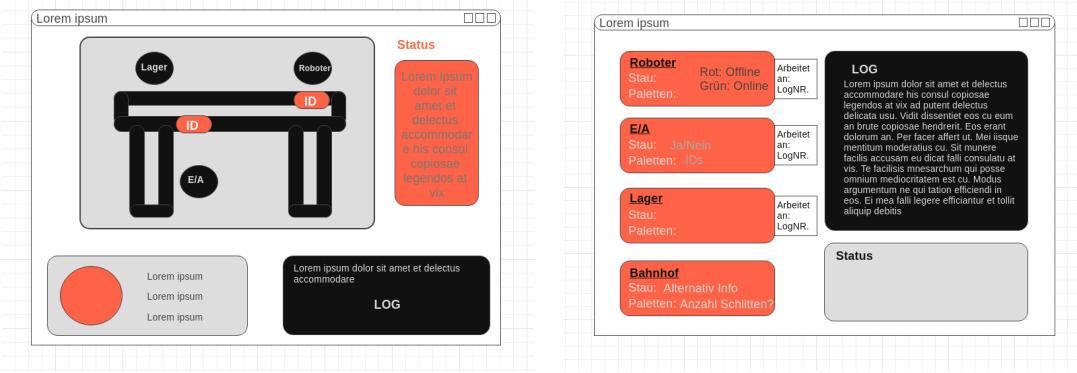
Die Oberfläche soll intuitiv verständlich sein, ohne dass der Anwender Hilfestellungen in Form einer Anleitung benötigt. Außerdem soll die Größe der Darstellung veränderbar sein.

4.1.2 Layout und Navigationsanforderungen

Die Gestaltung der Oberfläche ist den Entwicklern überlassen, diese erarbeiten Vorschläge, die dann mit Absprache der Auftraggeber übernommen werden.

Simulation FDZ	Praktikum Software Entwicklung
Anforderungsanalyse Spezifikation	26.03.2018

Mögliche Vorschläge:



Simulation FDZ	Praktikum Software Entwicklung
Anforderungsanalyse Spezifikation	26.03.2018

4.2 Schnittstellen zu externen Systemen und Geräten

4.2.1 Software Schnittstellen

Die einzige Software Spezifikation, auf die das System aufbaut, ist das Netzwerkprotokoll, das speziell für die Fabrik der Zukunft entwickelt wurde. Dieses Protokoll sendet an alle zuständigen Systeme die Befehle, die aktuell von der FDZ abgearbeitet werden.

4.2.2 Hardware Schnittstellen

Die Hardware Schnittstelle beschränkt sich lediglich auf dem Standard-Rechner, auf dem das System dann nach Fertigstellung laufen wird. Über dessen Netzwerkkarte empfängt das fertige System die Befehle, auf die es dann entsprechend reagiert und die Darstellung abändert.

4.2.3 Kommunikationsschnittstellen

(Siehe Spezifikation.pdf Kapitel 3.3)

(Siehe Implementierung.pdf Kapitel 3.1.1 – 3.1.2)

5. Vorhandenes System

(Siehe DocumentationNetworkCommands.pdf)

5.1 Befehle

(Siehe DocumentationNetworkCommands.pdf Kapitel 4.1.2 – 4.1.4)

5.2 Stationen

(Siehe Spezifikation.pdf Kapitel 1)

5.3 Schlitten / Palleten

(Siehe Spezifikation.pdf Kapitel 2.2, Spec.pdf Kapitel 1.2)

Simulation FDZ	Praktikum Software Entwicklung
Anforderungsanalyse Spezifikation	26.03.2018

5.4 Datenformat

(Siehe DocumentationNetworkCommands.pdf Kapitel 4.2)

6. System Randbedingungen

Das System muss in Java programmiert werden. Nach Abspreche mit dem Auftraggeber und Abgleichen mit den schon vorhandenen Systemen der Fabrik wurde die Java Version 1.8 für die Implementierung festgelegt. Die grafische Benutzeroberfläche des Systems soll in JavaFX implementiert werden.

Prototypen, Code und Dokumentation des Systemes müssen auf dem Repository des IISYS hochgeladen werden (Mercurial).

Es wurden keine weiteren Einschränkungen der Entwicklungstools festgelegt.

7. System Compilance

Das System wird zur Zweck Studienarbeit hergestellt und enthält keinen lizenzierten Inhalt.